

UNITED STATES PATENT APPLICATION

FOR

**METHOD AND APPARATUS FOR CONTROLLING AN ANIMATRONIC DEVICE USING  
A WEB ENABLED CELLULAR PHONE**

Inventor(s):        Brian Shuster  
                         Edwin Sahakian  
                         Craig James Beasley

Prepared by:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN  
12400 Wilshire Boulevard, 7<sup>th</sup> Floor  
Los Angeles, California 90025  
(310) 207-3800

002227 657260

## Field of the Invention

Description of Related Art

One current product that exists under the market name "Activemates", as marketed by Microsoft Corporation, operates with a computer and applicable software to allow a computer user to run programs (content) associated with an animatronic device. This software can send/play audio through the device, and also control or trigger control of the mouth on the device to animate it. However, this product provides feedback based on the user's actions locally, and does not offer control from another user, either directly or indirectly. Specifically, there is no provision for controlling the device over a network using a web enabled cell phone.

## SUMMARY OF THE INVENTION

- 1 -

received signals to an animatronic control command and send the animatronic control command to a server for sending to an addressed animatronic device. In one embodiment, the disclosed methods and apparatus determine if the animatronic control command is serviceable by the animatronic device. The animatronic control command is sent unaltered to a receiving client if the animatronic control command is serviceable by the animatronic device connected to the receiving client. The animatronic control command may be modified to a valid animatronic control command if the animatronic control command is not serviceable by the animatronic device.

#### **BRIEF DESCRIPTION OF THE DRAWINGS**

**Figure 1** is a system block diagram of one embodiment of a system in which the apparatus and method of the invention is used.

**Figure 2** illustrates an exemplary processor system or user computer system that may implement embodiments of the present invention.

**Figure 3** illustrates a block diagram detailing portions of one embodiment of the server 102 and client 106 of **Figure 1** configured in accordance with one embodiment of the present invention.

**Figure 3a** illustrates a block diagram detailing portions of one embodiment of the server/client 106a of **Figure 1** configured in accordance with one embodiment of the present invention.

**Figure 4** is a flowchart of a server utilized when a web-enabled phone or similar device is used as a source of animatronic control signals.

**Figure 5** is a flowchart of an embodiment showing the flow at a web-enabled phone generating commands according to the embodiment illustrated in **Figure 4**.

## **DETAILED DESCRIPTION**

### **Definitions**

As discussed herein, a "computer system" is a product including circuitry capable of processing data. The computer system may include, but is not limited to, general purpose computer systems (e.g., server, laptop, desktop, palmtop, personal electronic devices, such as personal digital assistants, cell phones, etc.), personal computers (PCs), hard copy equipment (e.g., printer, plotter, fax machine, etc.), and the like. In addition, a "communication link" or connection refers to the medium or channel of communication. The communication link or connection may include, but is not limited to, a telephone line, a modem connection, an Internet connection, an Integrated Services Digital Network ("ISDN") connection, an Asynchronous Transfer Mode (ATM) connection, a frame relay connection, an Ethernet connection, a coaxial connection, a fiber optic connection, satellite connections (e.g. Digital Satellite Services, etc.), wireless connections, radio frequency (RF) links, electromagnetic links, two way paging connections, etc., and combinations thereof.

### **System Overview**

A description of an exemplary system, which incorporates embodiments of the present invention, is hereinafter described.

The present invention relates to a method and apparatus for controlling animatronic devices over a network.

**Figure 1** illustrates a system 100 configured in accordance with one embodiment of the present invention, containing a server 102, a network 104, and at least one client 106 with a set of animatronic devices 108 and a server/client 106a with no connected animatronic device.

As further described below, server 102 includes various hardware components. This includes a processor, memory, and one or more network interface cards. In addition, server 102 may also include a variety of other hardware devices, including, but not limited to, storage devices (including floppy disk drives, hard disk drives, and optical disk drives), input devices (including a mouse or keyboard), and output devices (including displays and printers).

Each of the computer systems in set of clients 106 and server/client 106a may include a variety of hardware components that may be similar to server 102 or may be a device configured for a specific application such as a web enabled cell phone 107. In addition, each computer system may also be equipped with a data port for connection of one or more animatronic and other devices. In one embodiment, these include communications ports such as serial ports, which conform to the Institute of Electrical and Electronics Engineers (IEEE) RS-232 standard, IEEE-1284 parallel port standard, universal serial port (USB), or infra-red (IR) port for connection to set of animatronic devices 108.

Network 104 is a network that operates with a variety of communications protocols to allow client-to-client and client-to-server communications. In one embodiment, network 104 is a network such as the Internet, implementing transfer control protocol/internet protocol (TCP/IP).

Set of animatronic devices 108 can be made of various materials to form a toy device or the like. Each is controlled through the use of a respective computer 106 to which it is connected.

**Figure 2** illustrates an exemplary computer system 200 that implements embodiments of the present invention. The computer system 200 illustrates one embodiment of server 102

and set of clients 106 and server/client 106a (**Figure 1**), although other embodiments may be readily used.

Referring to **Figure 2**, the computer system 200 comprises a processor or a central processing unit (CPU) 204. The illustrated CPU 204 includes an Arithmetic Logic Unit (ALU) for performing computations, a collection of registers for temporary storage of data and instructions, and a control unit for controlling operation for the system 200. In one embodiment, the CPU 204 includes any one of the x86, Pentium™, Pentium II™, and Pentium Pro™ microprocessors as marketed by Intel Corporation, the K-6 microprocessor as marketed by AMD, or the 6x86MX microprocessor as marketed by Cyrix Corp. Further examples include the Alpha™ processor as marketed by Compaq Corporation, the 680X0 processor as marketed by Motorola; or the Power PC™ processor as marketed by IBM. In addition, any of a variety of other processors, including those from Sun Microsystems, MIPS, IBM, Motorola, NEC, Cyrix, AMD, Nexgen and others may be used for implementing CPU 204. The CPU 204 is not limited to microprocessor but may take on other forms such as microcontrollers, digital signal processors, reduced instruction set computers (RISC), application specific integrated circuits, and the like. Although shown with one CPU 204, computer system 200 may alternatively include multiple processing units.

The CPU 204 is coupled to a bus controller 212 by way of a CPU bus 208. The bus controller 212 includes a memory controller 216 integrated therein, though the memory controller 216 may be external to the bus controller 212. The memory controller 216 provides an interface for access by the CPU 204 or other devices to system memory 224 via memory bus 220. In one embodiment, the system memory 224 includes

04633.P005x2

synchronous dynamic random access memory (SDRAM). System memory 224 may optionally include any additional or alternative high speed memory device or memory circuitry. The bus controller 212 is coupled to a system bus 228 that may be a peripheral component interconnect (PCI) bus, Industry Standard Architecture (ISA) bus, etc. Coupled to the system bus 228 are a graphics controller, a graphics engine or a video controller 232, a mass storage device 252, a communication interface device 256, one or more input/output (I/O) devices 268<sub>1</sub>-268<sub>N</sub>, and an expansion bus controller 272. The video controller 232 is coupled to a video memory 236 (e.g., 8 Megabytes) and video BIOS 240, all of which may be integrated onto a single card or device, as designated by numeral 244. The video memory 236 is used to contain display data for displaying information on the display screen 248, and the video BIOS 240 includes code and video services for controlling the video controller 232. In another embodiment, the video controller 232 is coupled to the CPU 204 through an Advanced Graphics Port (AGP) bus.

The mass storage device 252 includes (but is not limited to) a hard disk, floppy disk, CD-ROM, DVD-ROM, tape, high density floppy, high capacity removable media, low capacity removable media, solid state memory device, and combinations thereof. The communication interface device 256 includes a network card, a modem interface, or a similar communications device for accessing network 264 via communications link 260. In addition, communications interface device 256 includes communication ports such as serial ports (e.g., IEEE RS-232), parallel ports (e.g., IEEE-1284), Universal Serial Bus (USB) ports, and infra-red (IR) ports or other wireless port.

The I/O devices 268-268<sub>N</sub> include a keyboard, mouse, audio/sound card, printer, and the like. The expansion bus

controller 272 is coupled to nonvolatile memory 275, which includes system firmware 276. The system firmware 276 includes system BIOS 82, which is for controlling, among other things, hardware devices in the computer system 200. The system firmware 276 also includes ROM 280 and flash (or EEPROM) 284. The expansion bus controller 272 is also coupled to expansion memory 288 having RAM, ROM, and/or flash memory (not shown). The system 200 may additionally include a memory module 290 that is coupled to the bus controller 212. In one embodiment, the memory module 290 comprises a ROM 292 and flash (or EEPROM) 294.

As is familiar to those skilled in the art, the computer system 200 further includes an operating system (OS) and at least one application program, which in one embodiment, are loaded into system memory 224 from mass storage device 252 and launched after POST. The OS may include any type of OS including, but not limited or restricted to, DOS, Windows<sup>TM</sup> (e.g., Windows 95<sup>TM</sup>, Windows 98<sup>TM</sup>, Windows NT<sup>TM</sup>, Windows ME<sup>TM</sup>), Unix, Linux, OS/2, Macintosh<sup>TM</sup> OS (e.g., OS/9), Xenix, etc. The operating system is a set of one or more programs which control the computer system's operation and the allocation of resources. The application program is a set of one or more software programs that perform a task desired by the user.

In accordance with the practices of persons skilled in the art of computer programming, the present invention is described below with reference to symbolic representations of operations that are performed by computer system 200, unless indicated otherwise. Such operations are sometimes referred to as being computer-executed. It will be appreciated that operations that are symbolically represented include the manipulation by CPU 204 of electrical signals representing data bits and the maintenance of data bits at memory

locations in system memory 224, as well as other processing of signals. The memory locations where data bits are maintained are physical locations that have particular electrical, magnetic, optical, or organic properties corresponding to the data bits.

When implemented in software, the elements of the present invention are essentially the code segments to perform the necessary tasks. The program or code segments can be stored in a processor readable medium or transmitted by a computer data signal embodied in a carrier wave over a transmission medium or communication link. The "processor readable medium" may include any medium that can store or transfer information. Examples of the processor readable medium include an electronic circuit, a semiconductor memory device, a ROM, a flash memory, an erasable ROM (EROM), a floppy diskette, a CD-ROM, DVD or other optical disc, a hard disk, a fiber optic medium, a radio frequency (RF) link, etc. The computer data signal may include any signal that can propagate over a transmission medium such as electronic network channels, optical fibers, air, electromagnetic, RF links, etc. The code segments may be downloaded via computer networks such as the Internet, Intranet, etc.

**Figure 3** is a functional block diagram of server 102, including an operating system 300, an http server 302, a set of interactive hypertext mark-up language (HTML) pages 304, a set of common gateway interface (CGI) scripts 306, an animatronic server 308, a database 310, and a conferencing application 312. In one embodiment, operating system 300 provides network services for such network protocols as TCP/IP.

- Server 102 provides the following capabilities:

- Basic network communication with multiple remote clients (e.g. set of clients 106) simultaneously.
- Performing authentication and authorization of individual clients to communicate with animatronic server 308 on server 102 and/or other animatronic clients on remote clients (e.g., set of clients 106) that are also simultaneously connected to server 102.
- Servicing HTTP requests from clients (e.g., set of clients 106) on the World Wide Web and communicating with animatronic control client browser plug-ins (e.g., animatronic plug-in 354).
- Facilitating multiple users to communicate in both group and private conversations while connected to server 102.

HTTP server 302 is server software such as the product marketed under the name of Netscape Enterprise Server by Netscape Corporation, or the product marketed under the name of Internet Information Server by Microsoft Corporation. HTTP server 302 is capable of communicating with HTTP web clients via the HTTP protocol. In this capacity, HTTP server 302 is able to serve web clients with data (HTML, application output data, etc.).

Animatronic server 308 communicates with animatronic clients (e.g., animatronic client 356) and services their requests for control of other animatronic devices found on the network. When animatronic server 308 is initiated, it takes control of a TCP/IP service port and listens for incoming connections from various clients over a network such as network 104. When animatronic server 308 sees an incoming connection, it then attempts to confirm the validity of the client and the data that is being sent to animatronic server 308. Once this has been established, animatronic server 308 may then start a bidirectional conversation using a custom

language that both the server and the client understand.

Animatronic server 308 provides the following capabilities:

- Provide information on animatronic devices (e.g., set of animatronic devices 108) with which it is able to communicate.
- Act as a gatekeeper between client applications (e.g., between client applications running on set of clients 106 and server/client 106a), thereby controlling access to the animatronic devices.
- Understand and provide control of animatronic devices no matter how sophisticated the device. For example, if a user were to attempt to move the mouth of an animatronic doll on a client system and that doll did not have a controllable mouth - animatronic server 308 may identify this and return what components were actually controllable on the doll. Animatronic server 308 may also be configured to alter control requests to allow the request to be serviced. For example, if the animatronic device from the example above has a moveable head, the command may be altered to move the head for any mouth movements.
- Communicate with scripts/plugin-ins that allow a web-server such as HTTP server 302 to access animatronic server 308, thereby allowing multiple users to access it through a web-client without having to install or use an animatronic client.

Set of CGI scripts 306 enable HTTP server 302 to communicate with animatronic server 308, thereby allowing a web developer to create web content capable of accessing various animatronic devices via a standard HTTP client. In the simplest form, set of CGI scripts 306 allows server 102 to request a list of active animatronic clients accessing the

server and whether they are currently communicating and/or available for access via server 102.

Database 310 contains data of various types and that, when combined, forms content that is served to users via the server 102. Database 310, in one embodiment, is actually a collection of databases containing information that may be accessed through a standard such as open database connectivity (ODBC) standard, as promoted by Microsoft Corporation. Information that may be contained in database 310 includes a listing of all animatronic devices and respective clients to which they are connected, the capabilities of the animatronic devices (e.g., what functions the animatronic devices have) and their status (e.g., whether they are currently accessible or not), and any information for any users that correspond to a particular animatronic device.

Conferencing server 312 communicates with remote conferencing clients (e.g., conferencing client 358) and with animatronic server 308 to provide conferencing capabilities between clients and control of animatronic devices by animatronic server 308 on those clients.

The functionality provided by HTTP server 302, animatronic server 308, and conferencing server 312 may be implemented over several computers. In addition, the services and data provided by database 310 may also be implemented on one or more servers. Using multiple servers provides scalability for both the applications using the data (e.g., the various servers-other than the database server), and the applications providing the data (e.g., the database servers).

**Figure 3** also illustrates client 106, which contains an operating system 350, including network drivers for such networks (such as TCP/IP), an HTTP client 352, an animatronic

client 356, an animatronic plug-in 354, and a conferencing client 358. In one embodiment, also provided is an animatronic device driver 360 which to the port (e.g., serial, USB) used to transmit commands to the animatronic device.

Referring to **Figure 3**, client 106 is capable of performing the following functions:

- Basic network communication with server 102 and with other remote clients as directed by server 102.
- Control of animatronic devices (e.g., animatronic device 108) connected to itself via an interface port and the processing of command sequences sent to it from server 102 (e.g., animatronic server 308) before converting this data into commands that it sends to the connected toy device. Also provided is communication with the conferencing software contained on server 102 with special command options to allow control of other toy devices connected to other clients through server 102.

Conferencing client 358 provides a program that interfaces with the multi-user-conferencing facilities of server 102. In addition, conferencing client 358 coordinates with animatronic client 356 to provide synchronized movement of any animatronic devices connected to client 106 with what is being displayed by conferencing client 358.

HTTP client 352 is a software program that works with HTTP, and is also known as a "browser" application. Similar products include the product marketed by Netscape Corporation under the name Netscape Navigator or the product marketed by Microsoft Corporation under the name Internet Explorer. Animatronic plug-in 354 is a plug-in application that is specifically written to interface with HTTP client 352 to provide an interface to animatronic client 356.

Animatronic device driver 360 provides a standard GUI based interface for applications so that an application will be able to open the driver using a Win32 CreateFile call, and then the application will be able to communicate using ReadFile, WriteFile, and DeviceIOControl calls.

The driver will allow reading from the device and writing to the device, using vendor specific control transfers on the device's control endpoint. These will be control transfers with data payloads.

Since this driver in a preferred embodiment works with a USB device, and communications with USB devices are asynchronous in nature, the driver will work asynchronously, using Driver Managed Queues.

The driver will be responsible for handling error conditions such as Stalls and other "hard" errors gracefully, by letting the application know that there was an error during request processing.

In one embodiment, server/client 106a instead of being a client 106 computer having an animatronic device connected to it, may be a computer functioning as a web server which receives signals over the Internet from a web-enabled cellular phone or similar device. This computer operates to process the received signals and convert them to commands of the type expected to be received by a client 106 with a connected animatronic device and sends these commands to a server 102 which sends the commands to a client 106 with a connected animatronic device.

Referring to Figures 1, 3, 3a and 4, a web-enabled phone 107 connects at step 401 to a web server/client 106a having functionality corresponding to animatronic client 356. In addition, the web server/client 106a would have a module 359

for receiving commands of the type generated by the keys of a web enabled cell phone which are mapped to corresponding animatronic commands. In particular, after connection, the web server/client 106a would send an html form to the device at step 403. This form would be displayed on the device and show a set of actions which the user may select by operation of keys on a keypad of the device. Upon selecting a desired action, which would then be transmitted to the web server/client 106a, at step 405 the web server/client 106a would obtain the selected action, convert it to an animatronic command via key mapping module 359 and animatronic client 356 and at step 407, send the command to a server 102 which would then send the command to the addressed client 106 with a connected animatronic device 108. At step 409, if the session had ended, the process would stop until another connection had been established or go back to step 405 to wait for another response from the web-enabled phone 107 or other device.

Figure 5 shows the corresponding flow at the web-enabled phone or similar device. At step 501, connection with a web server/client 106a is established. The web server would then send an html form which would be displayed on the device at step 503. The user would then make a selection, by for example, pressing an up or down button until a desired action is highlighted. Upon pressing a send button or equivalent key, the test 505 for a new request would be returned yes and the selected action would be sent to the web server/client 106a at step 507. Otherwise, a test 509 would be made to see if the session was over. If no, the device would wait for a new request from the user at step 505. The flow shown in Figure 5 would typically not require any programming specific to this function within the web-enabled phone, but instead

would rely upon built-in functionality available by the device's ability to receive and display an html form.

Although the present invention has been described in terms of certain preferred embodiments, other embodiments apparent to those of ordinary skill in the art are also within the scope of this invention. Accordingly, the scope of the invention is intended to be defined only by the claims that follow.